```c
//C program to search an element in an array.

#include<stdio.h>

int search(int arr[], int n, int x){
      for(int i=0; i<n; i++){
            if(arr[i]==x){
                return i;
                break;
            }
      }
return -1;
}

int main(){
    int n;
    printf("Enter the size of the array: ");
    scanf("%d", &n);
    int arr[n];
    printf("\nEnter the elements of the array: \n");
    for(int i=0; i<n; i++){
        scanf("%d", &arr[i]);
    }
    printf("\n");
    int x;
    printf("Enter the elemnt to be searched: \n");
    scanf("%d", &x);
    if(search(arr,n,x)>=0)
    printf("The element %d is at index %d",x, search(arr,n,x));
    else
    printf("Element not found");

return 0;
}
```

Output:

```
Enter the size of the array: 5

Enter the elements of the array:
1 2 3 4 5

Enter the elemnt to be searched:
3
The element 3 is at index 2
```

//C program to print the maximum and minimum element of an array.

```c
#include<stdio.h>

int maxElement(int arr[], int n){
      int max = 0;
      for(int i=1; i<n; i++){
            if(arr[i]>arr[max])
            max = i;
      }
return arr[max];
}

int minElement(int arr[], int n){
      int min = 0;
      for(int i=1; i<n; i++){
            if(arr[i]<arr[min])
            min = i;
      }
return arr[min];
}

int main(){
      int n;
    printf("Enter the size of the array: ");
    scanf("%d", &n);
    int arr[n];
    printf("\nEnter the elements of the array: \n");
    for(int i=0; i<n; i++){
        scanf("%d", &arr[i]);
    }
    printf("\n");
    printf("The elements of the array:\n");
    for(int i=0; i<n; i++){
    printf("%d\t", arr[i]);
    }
    printf("\n");
    printf("The maximum element is: %d", maxElement(arr,n));
    printf("\nThe minimum element is: %d", minElement(arr,n));
    printf("\n");

return 0;
}
```

Output:

```
Enter the size of the array: 5

Enter the elements of the array:
1 2 3 4 5

The elements of the array:
1       2       3       4       5
The maximum element is: 5
The minimum element is: 1
```

```c
//C program to reverse an array.

#include<stdio.h>

int main(){
      int n;
    printf("Enter the size of the array: ");
    scanf("%d", &n);
    int arr[n];
    printf("\nEnter the elements of the array: \n");
    for(int i=0; i<n; i++){
        scanf("%d", &arr[i]);
    }
    printf("\n");
    printf("The elements of the array before reversing:\n");
    for(int i=0; i<n; i++){
        printf(" %d ", arr[i]);
    }
    printf("\n");
    int low = 0, high = n-1;
    while(low<high){
        arr[low]^=arr[high];
        arr[high]^=arr[low];
        arr[low]^=arr[high];
        low++;
        high--;
    }
    printf("The elements of the array after reversing:\n");
    for(int i=0; i<n; i++){
        printf(" %d ", arr[i]);
    }
   printf("\n");

return 0;
}
```

Output:

```
Enter the size of the array: 5

Enter the elements of the array:
1 2 3 4 5

The elements of the array before reversing:
 1  2  3  4  5
The elements of the array after reversing:
 5  4  3  2  1
```

//C program to sort an array.

```c
#include<stdio.h>

int main(){
      int n;
    printf("Enter the size of the array: ");
    scanf("%d", &n);
    int arr[n];
    printf("\nEnter the elements of the array: \n");
    for(int i=0; i<n; i++){
        scanf("%d", &arr[i]);
    }
    printf("\n");
    printf("The elements of the array before sorting:\n");
    for(int i=0; i<n; i++){
        printf(" %d ", arr[i]);
    }
    printf("\n");
    for(int i=0; i<n; i++ ){
        for(int j=i+1; j<n; j++){
              if(arr[j]<arr[i]){
                  arr[i]^=arr[j];
                  arr[j]^=arr[i];
                  arr[i]^=arr[j];
              }
        }
    }
    printf("The elements of the array after sorting:\n");
    for(int i=0; i<n; i++){
        printf(" %d ", arr[i]);
    }
    printf("\n");
return 0;
}
```

Output:

```
Enter the size of the array: 5

Enter the elements of the array:
12 11 31 5 30

The elements of the array before sorting:
 12  11  31  5  30
The elements of the array after sorting:
 5  11  12  30  31
```

//C program to print multiplication of 2 matrices and print the transport of the resultant matrix.

```c
#include<stdio.h>

void transpose(int arr[3][3]);
void multMatrix(int a[3][3], int b[3][3]){
    int mul[3][3],i,j,k;
    for(i=0; i<3; i++){
          for(j=0; j<3; j++){
                mul[i][j]=0;
                for(k=0; k<3; k++){
                      mul[i][j] += a[i][k]*b[k][j];
                }
          }
    }
    printf("The multiplied matrix is: \n");
    for(i=0; i<3; i++){
          for(j=0; j<3; j++){
                printf("%d\t", mul[i][j]);
          }
          printf("\n");
    }
    transpose(mul);
}

void transpose(int arr[3][3]){
      int i,j;
    printf("The transposed matrix is: \n");
    for(i=0; i<3; i++){
          for(j=0; j<3; j++){
                printf("%d\t", arr[j][i]);
          }
          printf("\n");
    }
    printf("\n");
}
```

```c
int main(){
    int i,j;
    int a[3][3], b[3][3];
    printf("Enter the values of matrix a: \n");
    for(i=0; i<3; i++){
          for(j=0; j<3; j++){
                scanf("%d", &a[i][j]);
          }
    }
    printf("\n");
    printf("The matrix a is:\n");
    for(i=0; i<3; i++){
          for(j=0; j<3; j++){
                printf("%d\t", a[i][j]);
          }
          printf("\n");
    }

    printf("\n");
    printf("Enter the values of matrix b: \n");
    for(i=0; i<3; i++){
          for(j=0; j<3; j++){
                scanf("%d", &b[i][j]);
          }
    }
    printf("\n");
    printf("The matrix b is:\n");
        for(i=0; i<3; i++){
          for(j=0; j<3; j++){
                printf("%d\t", b[i][j]);
          }
          printf("\n");
    }
     printf("\n");
    multMatrix(a,b);

return 0;
}
```

Output:

```
Enter the values of matrix a:
1 2 3 4 5 6 7 8 9

The matrix a is:
1       2       3
4       5       6
7       8       9

Enter the values of matrix b:
9 8 7 6 5 4 3 2 1

The matrix b is:
9       8       7
6       5       4
3       2       1

The multiplied matrix is:
30      24      18
84      69      54
138     114     90

The transposed matrix is:
30      84      138
24      69      114
18      54      90
```